بسم الله الرحمن الرحيم

وأفضل الصلاة و أتم التسليم على رسولنا الكريم

السلام عليكم ورحمة الله وبركاته

رضا الناس غاية لا تدرك

ضحكت فقالوا ألا تحتشم

بكيت فقالوا ألا تبتسم

بسمت فقالوا يرائي بها

عبست فقالوا بدا ما كتم

صمّت فقالوا كليل اللسان

نطقت فقالوا كثير الكلم

حلمت فقالوا صنيع الجبان

ولو كان مقتدرا لانتقم

بسلت فقالوا لطيش به

وما كان مجترئاً لو حكم

يقولون شذ إذا قلت لا

وإمعة حين وافقتهم

فأيقنت أني مهما أرد

رضا الناس لا بد من أن أذم

تحية طيبة وبعد:

سنتناول في هذا الموضوع طريقة تسجيل البرنامج وتفعيله و الذي تم تناوله سابقا في الفيجوال بيسك ٦ ورغم الوقت بين الموضوعين لم أجد موضوع يطرح نفس الفكرة تماما وإنما بتشابه وبعض التعقيد لذلك سأعيد بناء الفكرة و أطرحها في هذا الموضوع إن شاء الله من جديد بأسلوب بسيط ومطول قليلا من الناحية النظرية ليتم استيعابها بالشكل الكافي .

بداية وقبل كل شيء الطريقة مطروحة لأسباب تعليمية وطبعا لن أناقش فعالية الحماية أو قوة الكسر لان الأمر مفروغ منه ولا أحب الولوج فيه لكن إن شاء الله سيتم تغطية بعض الجوانب الوقائية لهذا الموضوع .

بسم الله نبدأ:

سنعتمد في برنامجنا الذي سنقوم بتصميمه طريقة الحماية عبر تسجيل وتفعيل البرنامج وهي أن برنامجنا عند تشغيله يظهر للمستخدم رقم ويطالبه بإدخال الرقم المقابل له وهذا الرقم طبعا سيكون فريد أو مميز أي يختلف من جهاز إلى أخر ولنا الحرية في حال اعتماد رقم قطعة في الجهاز أو دالة معينة وبعد ظهور الرقم للمستخدم يتم إرساله لك لتوليد الرقم المقابل له و إرساله للعميل ليعمل البرنامج إذا كان الرقم مطلوب بمجرد التشغيل أو ليقوم بتفعيل كامل الميزات إذا كنت عرضته بطريقة تعطيل بعض النوافذ أو الخصائص.

إذا الفكرة الأساسية أن برنامجنا يطلب رقم ليصبح نسخة كاملة .

بداية نستخرج رقم فريد من جهاز الحاسوب بحيث يكون هذا الرقم يختلف بين كل حاسب و أخر كرقم القرص الصلب أو المعالج أو ... ونقوم بعمليات تمويه له لكي لا يظهر مكشوفا كما هو فينتج رقم التسجيل للمستخدم.

الآن نقوم بعلميات تمويه أخرى لرقم التسجيل لنشتق رقم التفعيل فالبرنامج يحوي رقم القطعة أي إحدى عتاد الحاسوب ورقم تسجيل ورقم تفعيل فرقم القطعة لا علاقة لنا به لأننا بعد استخرجاه عدلناه ليظهر بطريقة أخرى والناتج سيكون رقم التسجيل الآن نعيد نفس الكرة ونعدل التسجيل لاستخراج رقم التفعيل وبذلك برنامجنا يجوي الأرقام الثلاثة و الظاهر منها هو رقم التسجيل فقط لدى العميل.

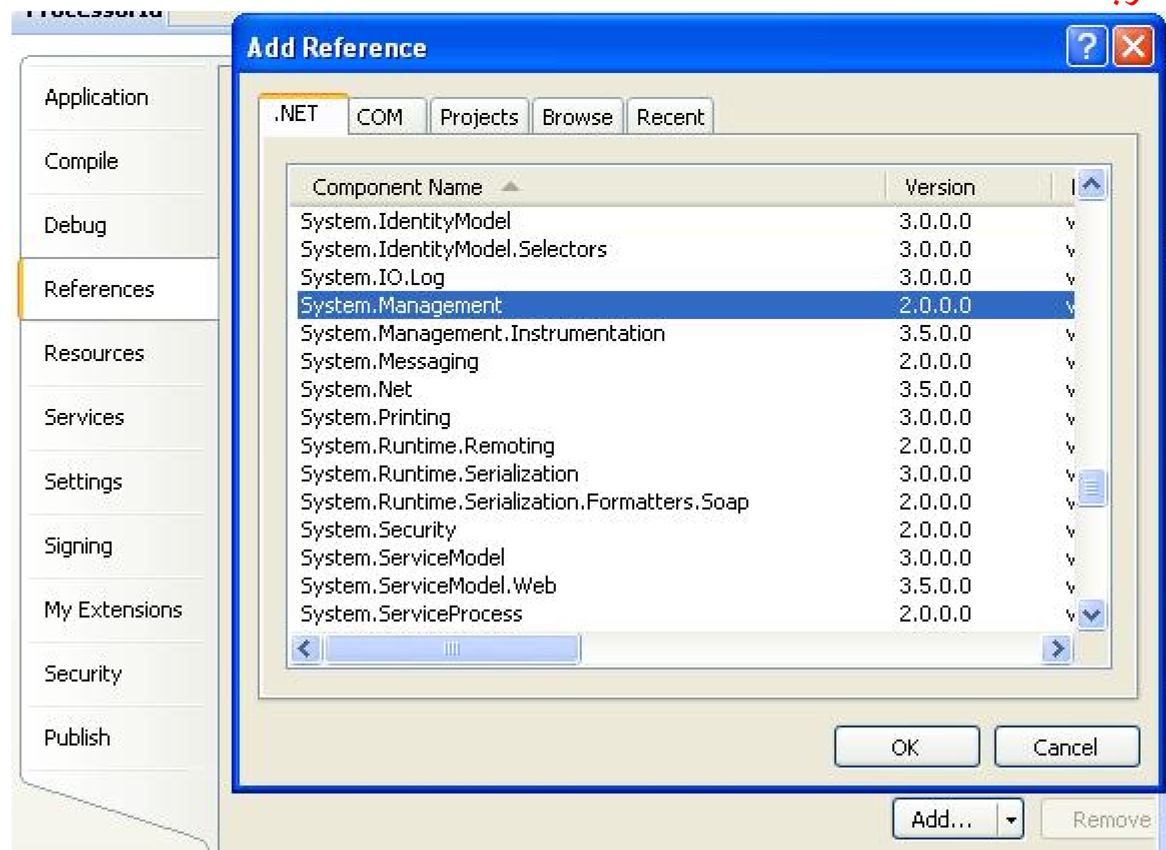
سيقوم العميل بإرسال رقم التسجيل لنا لنستخرج رقم التفعيل بنفس الطريقة التي استخدمناها في برنامجنا ونرسله له وعندها تتم المقارنة بين الرقمين وكسر البرنامج من قبل المهندس العكسي عفوا أقصد المقارنة بين الرقمين وتفعيل البرنامج في حال التساوي.

الطريقة التي سنتبعها في درسنا هي رقم المعالج البر وسيسور.

زمن vb6 كنا نستخدم مكتبة ميكروسوفت WbemScripting لاستخراج رقم المعالج وفي الدوت نت سنستخدم System.Management.dll لاستخراج الرقم و المكتبة نستطيع من خلالها استخراج مواصفات عتاد الجهاز لكم كبير من هذا العتاد ما يهمنا في موضوعنا هو رقم المعالج كونه فريد ونريد الاعتماد عليه في تغيير الرقم لكل مستخدم لبرنامجنا.

افتح Microsoft Visual Studio 2008 و أنشأ مشروع جديد وسمه ما شئت

توجه الى نافذة Solution Explorer واضغط عل المشروع بالزر الأيمن واختر الخصائص Properties ادخل إلى التبويب References و أضف Add References المكتبة System.Management كما في الصورة التالية:



بعد إضافة المكتبة توجه للنموذج أي الفورم و أضف صندوق نص تحت مسمى TextBoxId وزر أمر وسمه GetIdButton

التصميم :



الآن ضغطتين متتاليتين على زر الأمر لإدخال الكود المناسب لاستخراج رقم المعالج ووضعه في صندوق النص :

وبالأحرى هذا الكود كاملا أولا استيراد لوظائف المكتبة وبعدها كود الزر:

```
Imports System.Management
Public Class Form1
    Private Sub GetIdButton_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles GetIdButton.Click
        Dim Searcher As ManagementObjectSearcher
         Searcher = New ManagementObjectSearcher("Select ProcessorId From
Win32_Processor")
        For Each Device As ManagementObject In Searcher.Get
            For Each Prop As PropertyData In Device.Properties
                TextBoxId.Text = (Prop.Value.ToString)
```

```
        Next
    Next
End Sub

End Class
```

قم بتشغيل البرنامج و أضغط على الزر وستجد بإذن الله رقم المعالج يطبع في صندوق النص.

بالنسبة لشرح الكود السابق :

في البداية استيراد فضاء المكتبة وفي كود الزر قمنا بالبحث في مواصفات المعالج Processor عن قيمة Value الخاصية ProcessorId بواسطة حلقة For Each وعند إيجادها نستخرجها في صندوق النص.

ربما يسأل البعض لماذا لا نعرض الرقم مباشرة بدون الحلقة والبحث فيها فالجواب سيكون لان رقم الخاصية بين خصائص المعالج لن تكون الأولى وربما يختلف ترتيبها من جهاز لأخر لذلك نحن نبحث عنها ضمن المواصفات كلها وعند إيجادها يتم قنص القيمة و وضعها في صندوق النص.

الآن نستطيع استخدام رقم المعالج كما هو لعمليات التسجيل والتفعيل بحيث يكون الرقم مباشرة رقم تسجيل ورقم التفعيل يكون بإضافة بعض أرقام أو معادلة له لكن بهذه الطريقة إذا كان المستخدم لديه بعض البديهة فسينتبه إلى الرقم ويكتشف انه رقم المعالج وعندها يعرف التغير الذي حصل لرقم التفعيل بحذف رقم المعالج من السلسة لذلك سوف نقوم بتشفير الرقم و إظهاره بطريقة تجعل المستخدم لا يعرف ما هو الرقم الذي اعتمدنا عليه لذلك الآن نتجه لإنتاج رقم التسجيل والتفعيل .

أدرج صندوقي نص جديدين تحت مسميات TextBoxReg و TextBoxAct ليصبح التصميم:



الآن سنتجه لاستخراج رقم التسجيل والتفعيل عبر تحويل أو تشفير السلسة النصية التي تعبر عن رقم المعالج و التي تظهر في صندوق النص الأول TextBoxId سأستخدم دالة التحويل إلى الاسكي (الناتج سيكون رقمي) وهذه طرقة سهلة لتلقين رقم التفعيل للعميل كونها محصورة بالشكل الرقمي من ٠ إلى ٩ فقط وبعيدة عن الرموز أو الأحرف

أضف الدالة التالية لبرنامجك :

```
Function Str2Int(ByVal InStrng As Object) As String
    Dim StrLn As Integer
    Dim Cntr As Integer
    Dim NewStr As String
```

```
        Str2Int = ""
        StrLn = Len(InStrng)
        If StrLn = 0 Then Exit Function
        NewStr = ""
        For Cntr = 1 To StrLn
            Select Case Mid(InStrng, Cntr, 1)
                Case "0" To "z"
                    NewStr = NewStr & Asc(Mid(InStrng, Cntr, 1))
            End Select
        Next Cntr
        Str2Int = NewStr
    End Functio
```

و أضف في أخر كود الزر السطرين التالين :

```
        TextBoxReg.Text = Str2Int(TextBoxId.Text)
        TextBoxAct.Text = Str2Int(TextBoxReg.Text)
```

أي يصبح الكود كاملاً :

```
Imports System.Management
Public Class Form1
    Private Sub GetIdButton_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles GetIdButton.Click
        Dim Searcher As ManagementObjectSearcher
         Searcher = New ManagementObjectSearcher("Select ProcessorId From
Win32_Processor")
        For Each Device As ManagementObject In Searcher.Get
            For Each Prop As PropertyData In Device.Properties
                TextBoxId.Text = (Prop.Value.ToString)
            Next
        Next
        TextBoxReg.Text = Str2Int(TextBoxId.Text)
        TextBoxAct.Text = Str2Int(TextBoxReg.Text)
    End Sub
    Function Str2Int(ByVal InStrng As Object) As String
        Dim StrLn As Integer
        Dim Cntr As Integer
        Dim NewStr As String

        Str2Int = ""
        StrLn = Len(InStrng)
        If StrLn = 0 Then Exit Function
        NewStr = ""
        For Cntr = 1 To StrLn
            Select Case Mid(InStrng, Cntr, 1)
```

```
            Case "0" To "z"
                NewStr = NewStr & Asc(Mid(InStrng, Cntr, 1))
        End Select
    Next Cntr
    Str2Int = NewStr
End Function
End Class
```

شغل البرنامج وجرب الضغط على الزر وسيظهر أمامك الشكل التالي :



فالرقم الأول هو رقم المعالج سنخفيه عن العميل

الرقم الثاني هو رقم التسجيل مشتق بتحويل رقم المعالج إلى قيم الاسكي وهو سيظهر للعميل ويرسله لنا

الرقم الثالث رقم التفعيل مشتق من رقم التسجيل بنفس الطريقة الأولى وهو سيكون مخفي للعميل وهذا الرقم سيكون لدينا أيضا بالقيام بعملية تحويل التسجيل إلى اسكي أي قيم رقمية لإرساله إلى العميل وعندما يدخله تتم المقارنة بينه وبين رقم التفعيل المخفي عنه إذا تساوى الرقمين تتم العملية و إلا فلا

وهذا الشكل يوضع العملية:



فالمربعات ذات اللون الأسود لن تظهر للعميل و إنما سيظهر فقط رقم التسجيل الذي سيرسله لنا أي المربع الأبيض الأول ومربع يطلب منه إدخال رقم التفعيل المرسل له من قبلنا وهو المربع الأخير

وبعد إدخال الرقم تتم المقارنة بين الرقمين في المربعين الأخيرين في حال التساوي تتم العملية وإلا فلا

بالنسبة لطريقة التشفير فكما تلاحظون حاولنا استخدام دالة التحويل لقيم الاسكي للمفاتيح ويمكنكم اعتماد أي طريقة لتشفير الرقم حيث هناك طرق لا حصر لها يمكننا إجراء غربلة للرقم أو الاقتصاص منه أو الزيادة عليه أو تعديله بإضافة واحد أو أكثر أو تشويش السلسة كلها أو أخذ مواضع محددة من السلسة أو.............الخ

...

الأن طريقة المقارنة بين رقم التفعيل المخفي و الرقم الذي أدخله العميل

طبعا وجود المقارنة بحسب معلومات المهندس العكسي يسهل تجاوز البرنامج وكسره

الطريقة المحلية هي الأتي :

أضف زر أمر وسمه ButtonOk وصندوق النص الرابع الذي سيدخل العميل فيه الرقم سمه TextBoxUser

و أضف كود الزر ButtonOk التالي:

```
    Private Sub ButtonOk_Click(ByVal sender As System.Object, 
ByVal e As System.EventArgs) Handles ButtonOk.Click
        If TextBoxUser.Text = TextBoxAct.Text Then
            MsgBox("تم تفعيل البرنامج أشكرك")
        Else
            MsgBox("الرقم المدخل خاطئ")
        End If
    End Sub
```

الآن جرب لترى النتيجة :



نحن اعتمدنا إظهار رسالة للعميل تفيد بأن الرقم صح أو خطأ أما بالشكل العملي فالأمر سيكون بتفعيل فتح نوافذ واستخدام إمكانيات كالطباعة مثلا أو فتح البرنامج إذا كان التفعيل يظهر عند إقلاع البرنامج مباشرة لذلك نحن نستعيض بعبارة

```
            MsgBox("تم تفعيل البرنامج أشكرك")
```

بالإجراء المطلوب تنفيذه عند إدخال القيمة الصحيحة وإلا لن نظهر له شيء لان المهندس العكسي ينصحنا بذلك

```
If TextBoxUser.Text = TextBoxAct.Text Then
    MsgBox("تم تفعيل البرنامج شكراً لك")
End If
```

ومن نصائحه لنا هي أن لا نستخدم MsgBox ونحن أيضاً سنلتزم بأوامره فكما ذكرت سابقا الرسالة لإيضاح نجاح العملية أما الأصل فهو استخدام الإجراء المناسب كفتح البرنامج أو تفعيل الميزات المحجوبة

ربما يسأل البعض ما دخل المهندس العكسي في الموضع أقول أنه ربما يفكر في الكسر لذلك نحن نريده أن يكسب الجائزة بعرق جبينه وليس بسهولة لذلك سنجعل عملياتنا مخفية عليه بعض الشيء لنكسبه بعض الخبرة ويكون الكسر تم بتعبه وعرق جبينه و إلا فنحن نتحرى الخير في الناس وفي عملائنا أنهم لا يحاولوا كسر برامجنا ويتركونا نتسبب.

نتابع مع بعض النصائح : لا تجعل مكان التفعيل هو نفسه مكان المقارنة وإعطاء الصلاحيات

ما المقصود بذلك ؟

المقصود أحبتي : أننا عند تفعيل برنامجنا بشكله الصحيح فأنه لا يتم تخزين الرقم الخاطئ و يتم تخزين الرقم لأنه صحيح فقط أما تفعيل الميزات فتحتاج لمقارنة أخرى في فورم أخر بعيدا عن مكان التفعيل بشرح بسيط لو أدخل المستخدم رقم التفعيل الصحيح ستظهر رسالة تفيد بذلك وبنفس الوقت نخزن الرقم الصحيح في مكان معين (البرنامج- الريجستري-ملف) وعند إقلاع البرنامج يتم المقارنة مرة أخرى بنفس الطريقة لان البرنامج ربما يتم تجاوزه من خلال الهندسة العكسية ويتم تدمير قسم المقارنة لذلك المقارنة الأساسية تتم في مكان أخر وواجهة التفعيل هي فقط لتخزين الرقم الصحيح وعند إقلاع البرنامج تتم المقارنة بين رقم التفعيل والرقم المخزن من جديد صح يفعل الميزات وإلا فلا وهكذا نقوم بحشو برنامجنا بعشرة مقارنات ونضع عشرة مقارنات أخرى احتياط مثل بعد التشغيل الخامس للبرنامج في اليوم الأول من كل شهر وبعد شهرين بذلك نضمن أن قفزات الجمل الشرطية مهما حطم منها يبقى شيء مفعوله موجود

الهدف : بهذه الطريقة التمويهية نجعل أمر تجاوز المقارنات صعب بالتالي لن يكسر وطبعا هنا لن نسبية بطريقة القفز وتدمير المقارنات ماذا لو فكر بتدمير جميع المقارنات فالأمر صعب لان برنامجك ربما يحوي مقارنات عادية كالبحث وغيره بالتالي لن يعمل البرنامج بشكل صحيح فليس أمامه في طريقة القفز إلا العمل على البرنامج لمدة سنة كاملة وبعدين يعطينا نسخة مكسورة تعمل بطلاقة .

الطرقة الثانية للمهندس العكسي : هي استنتاج الرقم طبعا سيكون هذا الأمر مجدي له أكثر بخلاف العادة فبمجرد الوصول لأول مقارنة بين رقم التفعيل و الرقم المدخل سيكشفه إذا كان ذكي قليلا حسب معلوماتي المتواضعة عن هذه الأمور.

لكن هذه الطريقة تفتح أمامه مجالين :

الأول : أنه سيكشف الرقم للجهاز الذي يحاول الكسر عليه بالتالي يجب عليه الذهاب لكل جهاز والكسر عليه بهذه الحالة نضمن أن النسخ المكسورة ستكون أقل ما يمكن لأنه مش معقول يحمل العدة ويجوب على المستخدمين ياهو كسر برنامج فلان .

الثاني : هو الخطير لو استطاع كشف المعادلة بالتالي يستطيع عمل مولد أرقام مثل الذي لدينا وهذا سنحاول تجنبه بتعقيد المعادلة المخصصة لاستخراج رقم المعالج وتشفيره بحيث لا يستطيع فك عمليات التشفير على الرقم.

أتوقع كافي نظري حول تجنب الكسر و لا أقول لا أو انه صعب الكسر فالسوق أكبر دليل لكن كان سعيي لتوليد فكرة حول الأمر للجميع حتى يناقشوه في برامجهم ويستخدموه بالشكل الأمثل الذي يناسبهم.

نعود لكودنا في الزر طبعا الدالة يمكنك أن ترميها في موديول وتخلص من رؤيتها أمامك لأنها جاهزة ولن نعدل عليها وإنما سنستدعيها فقط لذلك أول شيء قم بقص الدالة و الصقها في الموديل .

الآن سنجعل رقم المعالج يظهر بمجرد فتح الفورم طبعا فقط ننقل الكود من حدث ضغط الزر GetIdButton_Click إلى تحميل الفورم Form1_Load ولسنا بحاجة لأكثر من صندوقين نص واحد تسجيل والأخر تفعيل والقيم الوسطى نخزنها ضمن متغيرات idp idp1 idp2 فيصبح الكود كالأتي :

```
Imports System.Management

Public Class Form1
    Dim idp As String ' رقم المعالج
    Dim idp1 As String ' رقم التسجيل
    Dim idp2 As String ' رقم التفعيل

    Private Sub Form1_Load(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.Load
        Dim Searcher As ManagementObjectSearcher
        Searcher = New ManagementObjectSearcher("Select
ProcessorId From Win32_Processor")
        For Each Device As ManagementObject In Searcher.Get
            For Each Prop As PropertyData In Device.Properties
                idp = (Prop.Value.ToString)
            Next
        Next
        idp1 = Str2Int(idp)
        idp2 = Str2Int(idp1)
        TextBoxReg.Text = idp1
    End Sub
    Private Sub ButtonOk_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles ButtonOk.Click
        If TextBoxUser.Text = idp2 Then
            MsgBox("تم تفعيل البرنامج أشكرا لك")
        End If
    End Sub
End Class
```

لاحظ أننا الآن بحاجة لصندوقي نص هما TextBoxReg يظهر فيه رقم التسجيل و TextBoxUser لإدخال رقم التفعيل من قبل العميل أما البقية والزر لا تلزم لان القيم صارت ضمن متغيرات ولا ننسى الدالة ألصقناها بالموديول من قبل فشكل نافذة التفعيل الآن هو الأتي :



لو انتبهت للكود فجملة If تناقش الحالة الصحيحة وإذا خطأ لن نقوم بأي عمل لأننا بحالة توليد مسج بوكس أنه خطأ نسهل العملية بحسب النصائح السابقة وكما ذكرت أنفا المسج بوكس يجب الاستغناء عنه لأنه أولا نقطة ضعف وثانيا لن يفيدنا تخيل انه المستخدم فعل البرنامج خلاص أغلق البرنامج وهو يشغله ويراه مفعل أو أغلق فورم التفعيل و خزن الرقم أو حتى خزن الرقم وبس. وما في ضرورة نلقمه إياها بالملعقة بيب مسج بوكس بيب تم التفعيل يا عميل خلص أغلق فورمه التفعيل ورح من هنا بيب شكرا لك هيه صار عندك نسخة كاملة هذا الحشو مش مهم .

طيب الآن ربما يسأل السائل ربما يستنتج العميل أو حبيبنا العكسي أن الرقم المستخرج هو رقم المعالج وبنظرة بسيطة بالمقارنة بين رقمي التسجيل و رقم المعالج سينتبه أن الثاني مشتق من الأول بالتحويل الرقمي كيف لتشابه أرقام في المعالج مع المقابل الرقمي في حال التكرار مثلا لدي يظهر

رقم المعالج:

0383F9FF000006B1

رقم التسجيل:

48515651705770704848484848546649

فربما يفهم العملية لذلك سنحاول جعله لا يفهم العملية بالقيام بتشويش السلسلة النصية ومن ثم تحويلها إلى رقم وبذلك حتى الرقمين المتشابهين سيكون بالتشويش لديها قيم مختلفة :

لذلك أول شيء اذهب للموديول و الصق الدالة التالية أسفل الدالة السابقة صاحبة الاسكي ليصبح الموديول كاملا كالأتي:

```
Module Module1
    ' الاسكي الى التحويل
    Function Str2Int(ByVal InStrng As Object) As String
        Dim StrLn As Integer
        Dim Cntr As Integer
        Dim NewStr As String

        Str2Int = ""
        StrLn = Len(InStrng)
        If StrLn = 0 Then Exit Function
        NewStr = ""
        For Cntr = 1 To StrLn
            Select Case Mid(InStrng, Cntr, 1)
                Case "0" To "z"
                    NewStr = NewStr & Asc(Mid(InStrng, Cntr, 1))
```

```
            End Select
        Next Cntr
        Str2Int = NewStr
    End Function

    'التشويش دالة
    Public Function Obfuscate(ByVal origText As String) As
String
        Dim textBytes As Byte() =
System.Text.Encoding.Unicode.GetBytes(origText)
        For counter As Integer = 0 To textBytes.Length - 1
            If (textBytes(counter) > 31) And (textBytes(counter)
< 127) Then
                textBytes(counter) += CByte(counter Mod 31 + 1)
                If (textBytes(counter) > 126) Then
textBytes(counter) -= CByte(95)
            End If
        Next counter
        Return System.Text.Encoding.Unicode.GetChars(textBytes)
    End Function

End Module
```

نشوش بالدالة بهذه الطريقة

```
Idp1 = Obfuscate(Idp)
```

الآن رقم المعالج هو

0383F9FF000006B1

بالتشويش بدالة Obfuscate يصبح

16=:ODSUACEGIQ_P

الآن نحوله إلى الرقمي

495461587968838565676971738819580

لاحظ من جديد هذا الرقم يختلف عن القيمة السابقة لأننا قمنا بتشويش رقم المعالج قبل تحويله إلى قيم الاسكي طبعا أنت تستطيع التفعيل بالقيم الحرفية لكن الرقمية تبقى أسهل للتلقين كما ذكرنا أول الموضوع

حاول المقارنة بين اللونين الأحمرين في رقم المعالج ورقم التسجيل لن تستنتج شيء لان التشويش قضى على هذا الأمر.

بينما كان الأمر جلي قبل التشويش.

لاحظ الفرق السابق فلا علاقة واضحة أبدا خمسة أصفار وكل شفرة لها مقابل مختلف هكذا يصبح الأمر مشفر أكثر

عد للنموذج وعدل الكود كالأتي :

```
Imports System.Management

Public Class Form1
    Dim Idp As String
    Dim Idp1 As String
    Dim idp2 As String

    Private Sub FrmActiv_Load(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles MyBase.Load
        'الرقم التسلسلي للمعالج
        Dim Searcher As ManagementObjectSearcher
        Searcher = New ManagementObjectSearcher("Select
ProcessorId From Win32_Processor")
        For Each Device As ManagementObject In Searcher.Get
            For Each Prop As PropertyData In Device.Properties
                '%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                'استخراج الرقم
                Idp = (Prop.Value.ToString)
                     Next
        Next
                '%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                'تشويش الرقم
                Idp = Obfuscate(Idp)
                'تحويله الى اسكي
                Idp = Str2Int(Idp)
                'رقم التسجيل
                TextBoxReg.Text = Idp
                '%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                'تشويش الرقم
                Idp1 = Obfuscate(Idp)
                'تحويله الى اسكي
                'رقم التفعيل
                Idp1 = Str2Int(Idp1)
    End Sub

    Private Sub ButtonOk_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles ButtonOk.Click
        'متغير جمالي
        idp2 = Idp1
        If TextBoxUser.Text = idp2 Then
            MsgBox("تم تفعيل البرنامج شكراً لك")
        End If
    End Sub
End Class
```

اتضح الأمر أكثر الآن فنحن نقوم باستخراج رقم المعالج وتشويشه وتحويله إلى أسكي فينتج رقم التسجيل ونطبعه للمستخدم في صندوق النص ونطالبه بالمقابل الذي هو عبارة عن رقم التسجيل مشوش ومحول إلى أسكي من جديد

وبعدها المقارنة في حدث الزر لن أكرر المسج بوكس بس للتوضيح حول المقارنة وليس للاستخدام.

الآن ربما تتململ من أن رقم المعالج ١٦ رقم ينتج عنها رقم تسجيل ٣٢ رقم ورقم التفعيل سيكون بالتالي ٦٤ رقم فهذا كثير مع أنني أفضل طول الرقم لكننا سنكتفي بطول رقم التسجيل ٣٢ رقم ونقوم بتصغير رقم التفعيل إلى ١٤ رقم مثلا وذلك بعملية اقتصاص من رقم التفعيل كالآتي وذلك بإضافة السطر التالي في أخر الفورم لوود (اقتصاص سلسلة نصية)

```
'رقم 14 اقتصاص
Idp1 = (Idp1.Substring(0, 14))
```

طبعا لك الحرية في اختيار طول الرقم بتغير الرقم ١٤ إلى أي رقم آخر لكن طول الرقم يصعب عملية الاشتقاق أصلا.

كذلك يمكنك اختيار نقطة البدء من أي محرف •

الآن يسأل السائل إذا قام العميل بالتفعيل فهل سأقول له مبروك فقط أما ماذا يحدث

نبدأ من جديد :

عند إدخال الرقم الصحيح سنقوم بتخزينه وأكرر لك الحرية في اختيار موضع التخزين ولا يهم ظهور الرقم لأنه صحيح ونحن من زودناه به لذلك سألجأ إلى Settings

نتجه إلى Solution Explorer ونضغط على المشروع بالزر الأيمن ونختار Properties ونذهب لتبويب Settings ونضيف إليه متغير جديد سأسميه nameuser وليس numact شوية تمويه يبدو التجنيد مؤثر بي كثير.

اسم المتغير

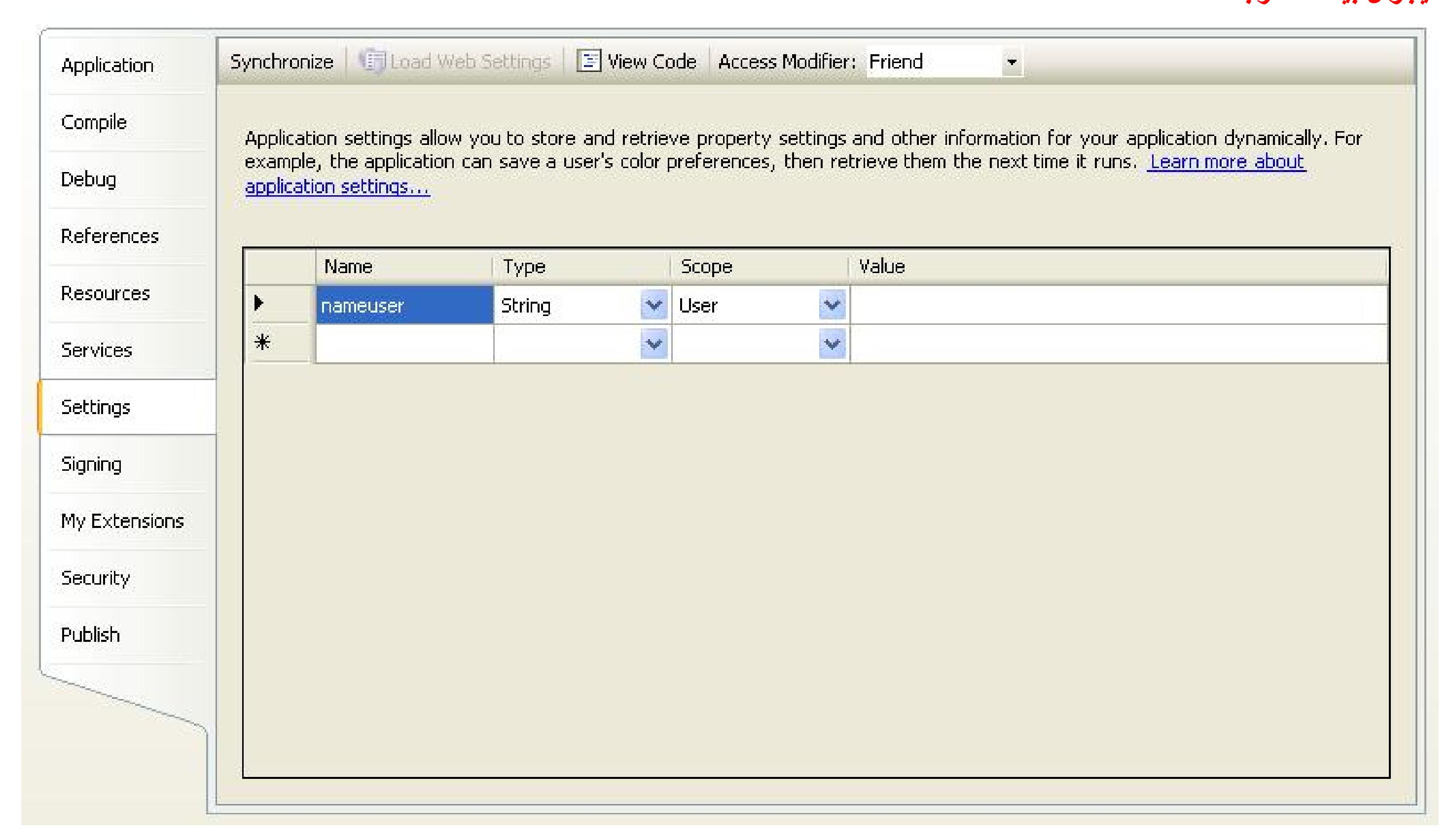
Name = nameuser

نوع المتغير

Type = String

قيمة المتغير

Value =



الآن سنتجه لكود الزر ونغيره إلى

```
    Private Sub ButtonOk_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles ButtonOk.Click
        'جما لي متغير
        idp2 = Idp1
        If TextBoxUser.Text = idp2 Then
            My.Settings.nameuser = TextBoxUser.Text
            My.Settings.Save()
            End
        End If
    End Sub
```

لاحظ الكود السابق في حال التساوي فإنه يسند الرقم إلى الإعدادات ثم يحفظها فقضية التفعيل كلها هنا هي هل الرقم متساوي إذا كان الجواب نعم فعندها لا يفعل الميزات بل يحفظ الرقم فقط وعملية التفعيل في أمكنة أخرى بعيدا عن مركز اختبار الرقم الظاهر للعميل و بذلك لن يظهر للعكسي الأمر واضح كالعادة .

هل تم الأمر لا خطوة أخيرة فقط :

الآن أصبح كود مشروعنا الكامل هو الأتي :

الموديول :

```
Module Module1
    ' الاسكي الى التحويل
    Function Str2Int(ByVal InStrng As Object) As String
        Dim StrLn As Integer
```

```
        Dim Cntr As Integer
        Dim NewStr As String

        Str2Int = ""
        StrLn = Len(InStrng)
        If StrLn = 0 Then Exit Function
        NewStr = ""
        For Cntr = 1 To StrLn
            Select Case Mid(InStrng, Cntr, 1)
                Case "0" To "z"
                    NewStr = NewStr & Asc(Mid(InStrng, Cntr, 1))
            End Select
        Next Cntr
        Str2Int = NewStr
    End Function

    'التشويش دالة
    Public Function Obfuscate(ByVal origText As String) As
String
        Dim textBytes As Byte() =
System.Text.Encoding.Unicode.GetBytes(origText)
        For counter As Integer = 0 To textBytes.Length - 1
            If (textBytes(counter) > 31) And (textBytes(counter)
< 127) Then
                textBytes(counter) += CByte(counter Mod 31 + 1)
                If (textBytes(counter) > 126) Then
textBytes(counter) -= CByte(95)
            End If
        Next counter
        Return System.Text.Encoding.Unicode.GetChars(textBytes)
    End Function

    End Module
```

الفورم:

```
Imports System.Management

Public Class Form1
    Dim Idp As String
    Dim Idp1 As String
    Dim idp2 As String

    Private Sub FrmActiv_Load(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles MyBase.Load
```

```
        'الرقم التسلسلي للمعالج
        Dim Searcher As ManagementObjectSearcher
        Searcher = New ManagementObjectSearcher("Select
ProcessorId From Win32_Processor")
        For Each Device As ManagementObject In Searcher.Get
            For Each Prop As PropertyData In Device.Properties
                '%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                'استخراج الرقم
                Idp = (Prop.Value.ToString)
            Next
        Next
        '%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        'تشويش الرقم
        Idp = Obfuscate(Idp)
        'تحويله الى اسكي
        Idp = Str2Int(Idp)
        'رقم التسجيل
        TextBoxReg.Text = Idp
        '%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        'تشويش الرقم
        Idp1 = Obfuscate(Idp)
        'تحويله الى اسكي
        'رقم التفعيل
        Idp1 = Str2Int(Idp1)
        'اقتصاص 14 رقم
        Idp1 = (Idp1.Substring(0, 14))
    End Sub

    Private Sub ButtonOk_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles ButtonOk.Click
        'متغير جمالي
        idp2 = Idp1
        If TextBoxUser.Text = idp2 Then
            'حفظ الاعدادت
            My.Settings.nameuser = TextBoxUser.Text
            My.Settings.Save()
            End
        End If
    End Sub
End Class
```

والتصميم التالي:



انتهى أمر التفعيل إذا خطأ لن يقوم بأي إجراء منعا لإجراء التجارب من قبل المعتدين و إذا صح يحفظ الرقم فقط

وتتم المقارنة لاحقاً.

الآن طريقة المقارنة بعيدا عن فورم التفعيل كالأتي

اذهب لأي فورم و ألصق كود الرقم

```
Imports System.Management

Public Class Form1
    Dim Idp As String
    Dim Idp1 As String
    Dim idp2 As String

    Private Sub FrmActiv_Load(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles MyBase.Load
        'الرقم التسلسلي للمعالج
        Dim Searcher As ManagementObjectSearcher
        Searcher = New ManagementObjectSearcher("Select
ProcessorId From Win32_Processor")
        For Each Device As ManagementObject In Searcher.Get
            For Each Prop As PropertyData In Device.Properties
                '%%%%%%%%%%%%%%%%%%%%%%%%%%%%
                'استخراج الرقم
                Idp = (Prop.Value.ToString)
```

```
            Next
        Next
        '%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        'الـرقـم تـشـويـش
        Idp = Obfuscate(Idp)
        'اسـكي الى تحـويـلـه
        Idp = Str2Int(Idp)
        'الـتـسـجـيـل رقـم
        TextBoxReg.Text = Idp
        '%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        'الـرقـم تـشـويـش
        Idp1 = Obfuscate(Idp)
        'اسـكي الى تحـويـلـه
        'الـتـفـعـيـل رقـم
        Idp1 = Str2Int(Idp1)
        'اقـتـصـاص 14 رقـم
        Idp1 = (Idp1.Substring(0, 14))
    End Sub
```

ففي أي فورم أو فورما محددة على مزاجك سيتم استنتاج الرقم كما تعلمنا سابقا في فورم التفعيل وحفظ الرقم في متغير

كما تلاحظ من الكود أخر قيمة كانت رقم التفعيل المكون من ١٤ رقم والمتغير هو Idp1

المقارنة كالتالي

```
        If My.Settings.nameuser = Idp1 Then
            'do thing
        Else
            'end
        End If
```

والآن ضع المقارنة أينما تريد في زر الطباعة في زر البحث في تحميل فورم المبيعات في .... الخ ....

واختر الإجراء المناسب سواء بإغلاق البرنامج أو بتعطيل بعض الأزرار أو بإخفاء نوافذ أو .... الخ.....

وكل ما كثرت المقارنات صار البرنامج أبطأ عفواً أقصد صار الأمر مبهم أكثر (ملح وفلفل)

عداك عند وضع مقارنات زمنية. (شهرية ... ساعية ... بعد ١٠ استعمالات ... الخ .....) بمحض إرادتك .

مداخلة :

حاول أيضا تفحص قيمة المتغير الأخير الذي سنخزن فيه رقم التفعيل بحيث إذا كانت فارغة الخروج من الإجراء وعدم التفعيل أو استخدم Try.

لأنه ربما بعد عشرين سنة يخرج نظام ويندوز لا يدعم المكتبة فيكون استخراج رقم المعالج خالي وتتم المقارنة مع قيمة خالية فتتساوى القيمة ويتم التفعيل أو يولد خطأ .

إلى هنا ننتهي من قسم التسجيل وننهب الآن لقسم التفعيل:

إن برنامج التفعيل يجب أن يتواجد لديك فقط أو للأشخاص الاعتباريين الذي يحق لهم توليد أرقام التفعيل بحسب رغبتك.

لان برنامج التفعيل هو الكيجن لبرنامجك إن صح التعبير

طريقة توليد أرقام التفعيل :

هنا الأمر بسيط جدا لأنه جزء صغير مما سبق

وهو جزء من مشروعنا فهو رقم التسجيل مشوش ومحول لاسكي ومقتص ١٤ رقم

ننشأ مشروع جديد ونضيف فيه موديول نضع فيه الدوال السابقة كما في التسجيل كالموديول السابق تماماً

وننشأ فورم جديد عليه زر أمر وصندوقي نص ButtonOk TextBoxReg TextBoxUser

التصميم :



كود زر التفعيل :

```
Public Class Form1
    Dim XXX As String
    Private Sub ButtonOk_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles ButtonOk.Click
        'التسجيل رقم
        XXX = TextBoxReg.Text
        '%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        'الرقم تشويش
        XXX = Obfuscate(XXX)
        'اسكي الى تحويله
        'التفعيل رقم
        XXX = Str2Int(XXX)
        'اقتصاص 14 رقم
        XXX = (XXX.Substring(0, 14))
        'التفعيل رقم استنتاج
        TextBoxUser.Text = XXX
    End Sub
End Class
```

طبعا لا حاجة للشرح فالأمر واضح مما سبق ومن الكود (فقط بعض التعديلات على رقم التسجيل)

الآن ما عليك إلا الضغط على زر رقم التفعيل بعد إدخال رقم التسجيل وسيظهر في الصندوق الثاني رقم التفعيل لترسله للعميل ويتمكن من تفعيل البرنامج بالشكل الصحيح .

وبذلك نكون قد ناقشنا الموضوع بطريقة مطولة بعض الشيء لذلك معذرة مني إليكم .

في المرفقات تجدون كل من :

مشروع التسجيل NourReg مشروع التفعيل NourAct ملفات تنفيذية للتجربة EXE

C#

حاولت طرح الموضوع للإخوة المستخدمين للسي شارب

بعد التعديل يصبح المثال كالتالي:

في الموديول WMI_Processor_Information.cs

الكود:

```
using System;
using System.Collections.Generic;
using System.Text;
using System.Management;

namespace WMI_ProcessorInformation
{
    public class WMI_Processor_Information
    {

        //Cpu serial,رقم المعالج
        public static string GetCpuId()
        {
            try
            {
                ManagementObjectSearcher searcher = new
ManagementObjectSearcher("root\\CIMV2",
                    "SELECT * FROM Win32_Processor");

                foreach (ManagementObject queryObj in searcher.Get())
                {
                    return  queryObj["ProcessorId"].ToString();
                }
            }
            catch (ManagementException e)
            {
                return null;
            }
            return null;
        }

        //التشويش//
        public static string Obfuscate(string origText)
        {
            byte[] bytes = Encoding.Unicode.GetBytes(origText);
            int num2 = bytes.Length - 1;
            for (int i = 0; i <= num2; i++)
            {
```

```
                if ((bytes[i] > 0x1f) & (bytes[i] < 0x7f))
                {
                    int index = i;
                    bytes[index] = (byte)(bytes[index] + ((byte)((i % 0x1f) + 1)));
                    if (bytes[i] > 0x7e)
                    {
                        index = i;
                        bytes[index] = (byte)(bytes[index] - 0x5f);
                    }
                }
            }
            return new string(Encoding.Unicode.GetChars(bytes));
        }
        //اسكي
       //('
    }
}
```

في الفورم

أضف الأدوات : Button1 TextBox1 TextBox2 TextBox3

وضع الكود التالي للزر ل Button1 :

```
        {
           //المعالج رقم
      textBox1.Text = (WMI_ProcessorInformation.WMI_Processor_Information.GetCpuId());
           //الرقم1 تشويش
            textBox2.Text = (WMI_Processor_Information.Obfuscate(textBox1.Text));
            //الرقم2 تشويش
            textBox3.Text = (WMI_Processor_Information.Obfuscate(textBox2.Text));

        }
```

يمكنكم إكماله كما تريدون انطلاقاً مما سبق في شرحنا السابق

وفي الختام:

- الغرض من برنامجنا هو غرض تعليمي.
- إذا كان برنامجك يستحق الحماية فقم بحمايته.
- لا تعرض برنامجك مباشرة و إنما قم بنشر نسخة محدودة المهام أصلا أي البرمجة غير كاملة بمثال عدم وجود البحث أبدا أو الطباعة أو .... وعند الطلب أرسل للعميل فقط نسخة كاملة تحتاج للتفعيل بإحدى الطرق ويفضل أن تتوسع في البحث عن حمايات أقوى مما قدمناه.
- ولكم مني جزيل الشكر لإصغائكم.

سبحانك اللهم وبحمدك أشهد أن لا إله إلا أنت أستغفرك و أتوب إليك

و لا حول ولا قوة إلا بالله العظيم

والسلام عليكم ورحمة الله وبركاته

أخوكم في الله نور نبهان (المزمجر سابقا)